

CLAIMS:

Having thus described our invention, what we claim as new, and desire to secure by Letters Patent is:

1 1. A method for merging two rulesets provided in rule-based systems associated with
2 originating applications executing at different locations, each ruleset comprising rules
3 in potential conflict with each other, and each ruleset being in a different rule format,
4 said method comprising:

5
6 a) communicating said rulesets to be merged over a distributed network to
7 an assimilator service device for receiving each said ruleset;

8
9 b) providing a merge policy to said assimilator device, said merge policy
10 comprising a set of specifications of partially-ordered priorities and/or mutual-
11 exclusion constraints;

12
13 c) translating said rulesets into a common core representation capable of
14 being implemented in any logic program rule engine provided in a rule-based
15 application at any location;

16
17 d) assimilating said rulesets to produce a new merged ruleset comprising
18

19 logic required for resolving potential conflicts among rules in accordance with said
20 merge policy, where said new merged ruleset is in a common core representation
21 capable of being implemented in any logic program rule engine provided in a rule-
22 based application at any location;

23

24 e) translating said new merged ruleset into one of said originating
25 application's said rule format; and

26

27 f) communicating said translated new merged ruleset over said distributed
28 network to the one of said originating applications.

1 2. The method according to Claim 1, wherein said assimilator device is employed to
2 merge rulesets in two or more rule formats from two or more originating applications
3 and communicate the translated new merged ruleset to one of said originating
4 applications.

1 3. The method according to Claim 1, wherein said assimilator device is employed for
2 updating rules included in a first ruleset imported from a rules-editor device.

1 4. The method according to Claim 2, wherein said assimilating step includes applying
2 one or more logic mechanisms in said merge policy for identifying conflicts and
3 resolving conflicts among said rules.

1 5. The method according to Claim 2, wherein a logic mechanism includes a priority
2 specification for expressing conflict resolution.

1 6. The method according to Claim 2, wherein a logic mechanism includes mutual
2 exclusion constraints.

1 7. The method according to Claim 2, wherein said core representation includes a
2 courteous logic program.

1 8. The method according to Claim 2, wherein said distributed network is the Internet.

1 9. An assimilator system for merging two or more rulesets provided in rule-based
2 systems associated with originating applications executing at different locations, each
3 ruleset having rules in potential conflict with each other, said system comprising:

4
5 a communications network enabling the transmission and receipt of rulesets to be
6 merged between said different locations;

7
8 a translator mechanism for translating each said ruleset from its rule format into a
9 common core representation capable of being implemented in any logic program rule
10 engine provided in a rule-based application at any location and for translating from
11 said common core representation into each said originating application's rule format;

12
13 a conflict transformer mechanism for receiving each said ruleset and assimilating said
14 rulesets to produce a new merged ruleset in accordance with a merge policy, said new
15 merged ruleset comprising specification of a set of partially-ordered priorities and/or
16 mutual-exclusion constraints that comprise logic required for resolving potential
17 conflicts among rules; and,
18

19 device for translating said new merged ruleset into a common core representation
20 capable of being implemented in any logic program rule engine provided in a rule-
21 based application at any location.

1 10. The assimilator system as claimed in Claim 9, wherein said new merged ruleset is
2 produced in said common core representation, said transforming device converting
3 said new merged ruleset into one of said originating formats.

1 11. The assimilator system as claimed in Claim 9, wherein said merge policy includes
2 one or more logic mechanisms for identifying and resolving conflicts among said
3 rules.

1 12. The assimilator system as claimed in Claim 11, wherein a logic mechanism
2 includes a priority specification for expressing conflict resolution.

1 13. The assimilator system as claimed in Claim 12, wherein a logic mechanism
2 includes mutual exclusion constraints for expressing conflict resolution.

1 14. The assimilator system as claimed in Claim 9, wherein said communications
2 network includes the Internet.

1 15. A program storage device readable by machine, tangibly embodying a program of
2 instructions executable by the machine to perform method steps for merging two
3 rulesets provided in rule-based systems associated with originating applications
4 executing at different locations, each ruleset comprising rules in potential conflict with
5 each other, and each ruleset being in a different rule format, said method comprising:

6
7 a) communicating said rulesets to be merged over a distributed network to an
8 assimilator service device for receiving each said ruleset;

9 b) providing a merge policy to said assimilator device, said merge policy
10 comprising a set of specifications of partially-ordered priorities and/or mutual-
11 exclusion constraints;

12
13 c) translating said rulesets into a common core representation capable of being
14 implemented in any logic program rule engine provided in a rule-based application at
15 any location.

16
17 d) assimilating said rulesets to produce a new merged ruleset comprising logic
18 required for resolving potential conflicts among rules in accordance with said merge
19 policy, where said new merged ruleset is in a common core representation capable of
20 being implemented in any logic program rule engine provided in a rule-based
21 application at any location;

22
23 e) translating said new merged ruleset into one of said originating application's
24 rule format; and

25
26 f) communicating said translated new merged ruleset over said distributed
27 network to the one of originating applications.

1 16. The program storage device readable by machine as claimed in Claim 15, wherein
2 said assimilator device is employed for updating rules included in a first ruleset
3 imported from a rules-editor device.

1 17. The program storage device readable by machine as claimed in Claim 15, wherein
2 after said assimilating step, a step of transforming said new merged ruleset from said
3 common core representation back to an originating format.

- 1 18. The program storage device readable by machine as claimed in 15, wherein said
- 2 assimilating step includes applying one or more logic mechanisms in said merge
- 3 policy for identifying conflicts and resolving conflicts among said rules.

TO: 2000 E469T660